**CONCORDIA UNIVERSITY**
**GINA CODY SCHOOL OF ENGINEERING AND COMPUTER SCIENCE**

**DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING**
**AERO483 (3 credits) - Final Project**

**Date issued: March 23, 2020** **Report Due Date: April 17, 2020**

---

**In your final report you must include the following:**

1. **A statement of any assumptions you have made and why you have made them.**
2. **Matlab plots showing the trajectory of the vehicle and the estimated trajectory.**
3. **Your Matlab .m files commented with headers indicating how each function is used**
4. **Any Simulink block diagrams and simulink files you generated.**
5. **Please keep the length of your report to within 5 pages. You can include as many figures and code you want in an appendix.**
6. **A signed Expectation of Originality form is required. Recall that plagiarism is a serious violation of the academic code of conduct.**
   provost.concordia.ca/academicintegrity/plagiarism

---

**Project Description**

For the following model of a quadcopter drone design and implement a Kalman filter to estimate the trajectory based on a given set of IMU and GPS data as described below.

The following model is of a quadcopter performing trajectories in the x-y plane at a constant height, as a function of a given set of input attitude angles (see figure 1), measured with an IMU.

The following kinematics model is used for quadcopter motion in the horizontal plane:

$$\dot{x}(t) = v_x(t) \tag{1}$$
$$\dot{y}(t) = v_y(t) \tag{2}$$
$$\dot{v}_x = g\tan\big(\theta(t)\big) + w_{v,x}(t) \tag{3}$$
$$\dot{v}_y = g\tan\big(\phi(t)\big) + w_{v,y}(t) \tag{4}$$

where $g$ is the gravitational acceleration of 9.8 m/s$^2$ and $w_{v,x}(t)$ and $w_{v,y}(t)$ are zero mean white noise both with standard deviation $\sigma_v(t)$ accounting for additional accelerations due to wind.

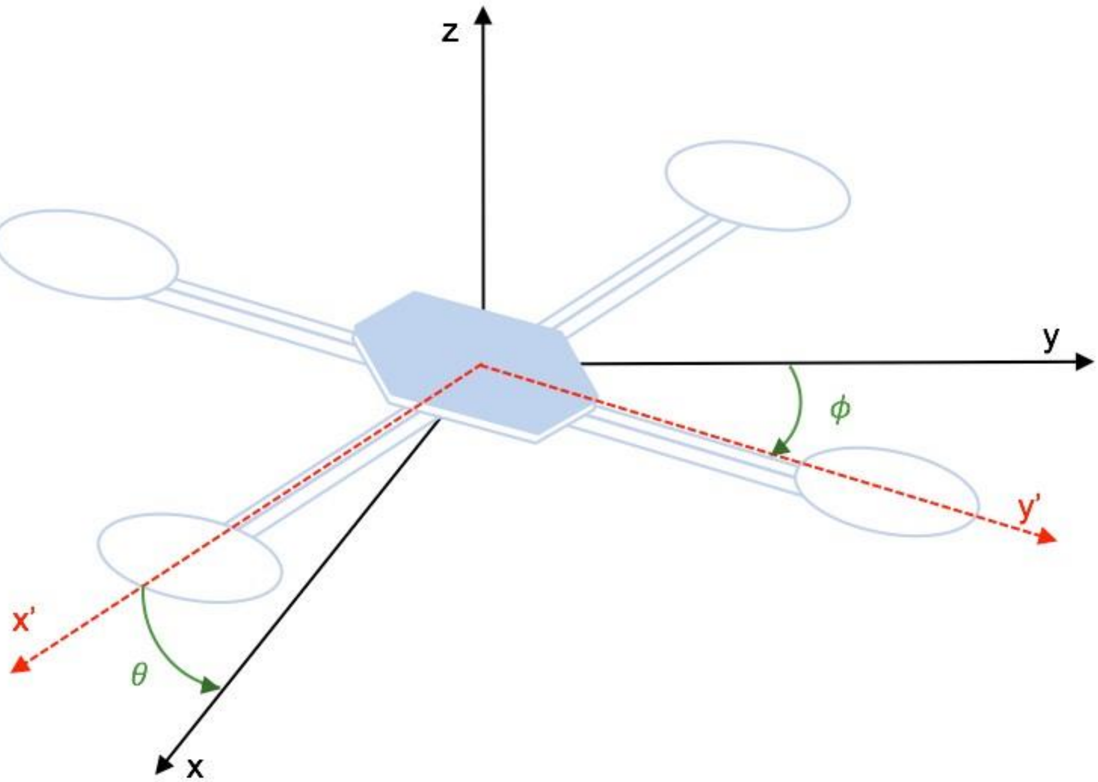


Figure 1: Quadcopter reference frames and attitude angles. Note that θ is defined as a rotation about -x (i.e. negative x), and ϕ is a rotation about -y (i.e. negative y). The particular θ shown in the example in the figure is negative, and the particular ϕ shown in the figure is positive. [picture courtesy of Bruno Carvalho]

Your program should be able to read the provided INS and GPS data by parsing as input the matrices ***ins*** and ***gps***. The first, second, and third columns of the ***ins*** matrix contain the measurement time, the measured pitch angle $\theta(t)$, and measured roll angle $\phi(t)$, respectively. Similarly, in the matrix ***gps*** the first column represents the measurement time, and the second and third columns contain the measured position coordinates $x$ and $y$, respectively. In all cases, each row of the matrix represents a unique measurement. [HINT: use load -ascii ins.txt]

For the quadcopter use the data provided. $\sigma_v = 0.03\ ms^{-2}$, $x(0) = 0$ m, $y(0) = 0$ m, $v_x(0) = 0$ ms$^{-1}$, $v_y(0) = 0$ ms$^{-1}$. $\mathbf{P}(0) = \mathbf{0}$. For the GPS measurements, assume a measurement error of

$$\boldsymbol{R} = \sigma_{GPS}^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

where $\sigma_{GPS} = 0.3\ m$

**Step 1 – System Simulation:** Build a simulation model in Matlab/Simulink for the quadcopter kinematics model and answer the following questions:

**Question 1:** Plot $\theta(t)$ and $\phi(t)$ on the same graph. How is $\theta(t)$ related to x(t), and how is $\phi(t)$ related to y(t)? What shape (in x-y space) is the trajectory that the operator has tried to command?

**Question 2:** Simulate the system dynamics, at an update period of 0.05 s, based on the INS mechanization equations, (i.e. based exclusively on the provided ***ins*** inputs). Plot the resulting x-y trajectory of the quadcopter. Also plot the provided ***gps*** x-y points on the same graph.

**Question 3:** Now simulate the system dynamics, also at an update period of 0.05 s, based on both the provided ***ins*** inputs plus random noise added to the simulation. Sample values from $w_{v,x}(t)$ and $w_{v,y}(t)$ at each timestep [HINT: use normrnd(0,0.03)] and include these sampled values in equations 3 and 4. Run the randomized simulation 10 separate times (you should get a different trajectory each time) and plot all 10 of these trajectories on the same graph.
In which portion(s) of the trajectory, if any, do the randomized trajectories track very relatively closely to one another? In which portion(s) of the trajectory, if any, do the randomized trajectories diverge from one another?

**Step 2 – Kalman Filter Design and Validation:** Design a Kalman Filter to integrate the mechanization equations of the INS with the GPS measurements and answer the following questions:

**Question 4:** Plot the Kalman Filter state estimate. On the same graph, also plot the separate INS mechanization trajectory and GPS positions from Question 2. Based on how closely the state estimate aligns with the INS mechanization trajectory vs GPS positions, what can you say about the relative magnitudes of **P** and **R** along each of the 4 major portions of the trajectory?